

BAB II

LANDASAN TEORI

2.1 Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan (JST) adalah suatu sistem pemrosesan informasi yang memiliki karakteristik unjuk kerja tertentu yang menyerupai jaringan syaraf biologis (Fausett, 1994). Jaringan Syaraf Tiruan merupakan salah satu sistem pemrosesan informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah dengan melakukan proses pembelajaran melalui perubahan bobot sinapsisnya. Pada proses pembelajaran, jaringan saraf tiruan akandimasukkan pola-pola *input* atau *output* lalu jaringan akan diajari untuk memberikan jawaban yang bisa diterima. Dalam Jaringan Syaraf Tiruan terdapat beberapa istilah yang sering muncul, yaitu:

1. *Neuron* atau *node* atau unit: sel syaraf tiruan yang merupakan elemen pengolahan jaringan syaraf tiruan. Setiap *neuron* menerima data *input*, memproses *input* tersebut kemudian mengirimkan hasilnya berupa sebuah *output*.
2. Jaringan: kumpulan *neuron* yang saling terhubung dan membentuk lapisan.
3. *Input*: sebuah nilai *input* yang akan diproses menjadi nilai *output*.
4. *Output*: solusi dari nilai *input*.
5. Bobot: nilai matematis dari sebuah koneksi antar-*neuron*.
6. Fungsi aktivasi: fungsi yang digunakan untuk meng-*update* nilai-nilai bobot per-iterasi dari semua nilai *input*. Fungsi aktivasi sederhana adalah mengalikan *input* dengan bobotnya dan kemudian menjumlahkannya (disebut penjumlahan sigma) berbentuk linier atau tidak linier dan sigmoid.
7. Paradigma pembelajaran: bentuk pembelajaran, *supervised learning*, atau *unsupervised learning*.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Seperti jaringan otak manusia, jaringan syaraf tiruan juga terdiri dari beberapa neuron. Neuron merupakan unit pemrosesan informasi yang menjadi dasar dalam pengoperasian jaringan syaraf tiruan. Terdapat tiga elemen pembentuk neuron, yaitu:

1. Himpunan unit-unit yang dihubungkan dengan jalur koneksi. Jalur-jalur tersebut memiliki bobot/kekuatan yang berbeda-beda. Bobot yang bernilai positif akan memperkuat sinyal dan yang bernilai negatif akan memperlemah sinyal yang dibawanya.
2. Suatu unit penjumlahan yang akan menjumlahkan input-input sinyal yang sudah dikalikan dengan bobotnya.
3. Fungsi aktivasi yang akan menentukan apakah sinyal dari input neuron akan diteruskan ke neuron lain atau tidak.

Jaringan Syaraf Tiruan juga memiliki beberapa kelebihan, antara lain:

1. Mampu mengakuisisi pengetahuan walau tidak ada kepastian.
2. Mampu melakukan generalisasi dan ekstraksi dari suatu pola data tertentu .
3. JST dapat menciptakan suatu pola pengetahuan melalui pengaturan diri atau kemampuan belajar (*self organizing*).
4. Memiliki fault tolerance, gangguan dapat dianggap sebagai noise saja.
5. Kemampuan perhitungan secara paralel sehingga proses lebih singkat.

Dilihat dari kemampuan Jaringan Syaraf Tiruan yang cukup baik, Jaringan Syaraf Tiruan bisa diterapkan pada:

1. Klasifikasi, memilih suatu input data ke dalam kategori tertentu yang sudah ditetapkan.
2. Asosiasi, menggambarkan suatu obyek secara keseluruhan hanya dengan bagian dari obyek lain
3. *Self organizing*, kemampuan mengolah data-data input tanpa harus mempunyai data target
4. Optimasi: menemukan jawaban terbaik sehingga dapat meminimalisasi fungsi biaya

2.1.1 Arsitektur Jaringan Saraf Tiruan

Pada mulanya Jaringan Syaraf Tiruan memiliki dua lapisan yang berupa *Input Layer* dan *Output Layer*, seiring perkembangannya terdapat satu lapisan lagi yaitu lapisan *Hidden Layer* lagi yang terletak di antara kedua lapisan sebelumnya. (Heaton, 2008 dalam Fanindia, dkk 2013):

1. Lapisan Masukan (*input layer*)

Input layer berisi *node-node* yang masing-masing menyimpan sebuah nilai masukan yang tidak berubah pada fase latih dan hanya bisa berubah jika diberikan nilai masukan baru. *Node* pada lapisan ini tergantung pada banyaknya *input* dari suatu pola.

2. Lapisan Tersembunyi (*hidden layer*)

Lapisan ini disebut *hidden layer* karena lapisan ini tidak pernah muncul. Akan tetapi semua proses pada fase pelatihan dan fase pengenalan dijalankan di lapisan ini. Jumlah lapisan ini tergantung dari arsitektur yang akan dirancang, tetapi pada umumnya terdiri dari satu lapisan *hidden layer*.

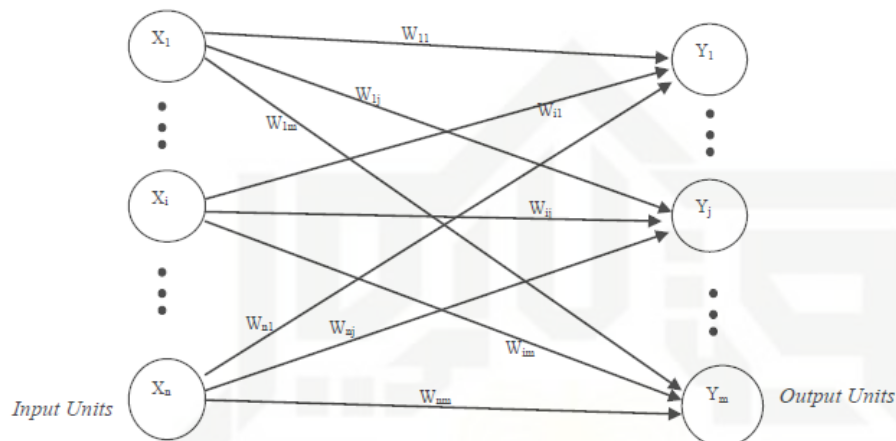
3. Lapisan Keluaran (*output layer*)

Output layer berfungsi untuk menampilkan hasil perhitungan sistem oleh fungsi aktivasi pada lapisan *hidden layer* berdasarkan *input* yang diterima.

Arsitektur jaringan berguna untuk menjelaskan arah perjalanan sinyal atau data pada jaringan. Algoritma pembelajaran akan memaparkan bagaimana bobot koneksi harus diubah agar pasangan masukan-keluaran yang diinginkan dapat tercapai. Dalam setiap perubahan harga bobot koneksi dapat dilakukan dengan berbagai cara, tergantung pada jenis algoritma pelatihan yang digunakan. Dengan mengatur besarnya nilai bobot ini diharapkan bahwa kinerja jaringan dalam mempelajari berbagai macam pola yang dinyatakan oleh setiap pasangan masukan-keluaran akan meningkat. Berikut 3 model arsitektur jaringan syaraf tiruan:

2.1.1.1 Jaringan Lapisan Tunggal (*Single Layer Network*)

Dalam jaringan ini, sekumpulan *input neuron* dihubungkan langsung dengan sekumpulan *output*-nya. Contoh JST yang menggunakan jaringan lapisan tunggal: ADALINE, Hopfield, Perceptron.



Gambar 2.1 Jaringan Lapisan Tunggal (Ratnaningtiyas, 2013)

Keterangan:

$x_{1,,}$: Nilai *input* ke 1 sampai n

$y_{1,,}$: Nilai *output* hasil pembangkitan nilai *input* oleh suatu fungsi aktivasi

w : Bobot atau nilai

Pada Gambar 2.1 diperlihatkan bahwa arsitektur jaringan *layer* tunggal dengan n buah masukan (x_1, x_2, x_n) dan m buah keluaran (y_1, y_2, y_m). Dalam jaringan ini semua unit *input* dihubungkan dengan semua unit *output*. Tidak ada unit *input* yang dihubungkan dengan unit *input* lainnya dan unit *output*-pun demikian.

2.1.1.2 Jaringan Lapisan Banyak (*Multilayer Network*)

Jaringan ini memiliki 3 jenis lapisan, yaitu lapisan input, lapisan tersembunyi, dan lapisan output. Contoh JST menggunakan jaringan lapisan banyak: MADALINE, Backpropagation, dan Neocognitron.

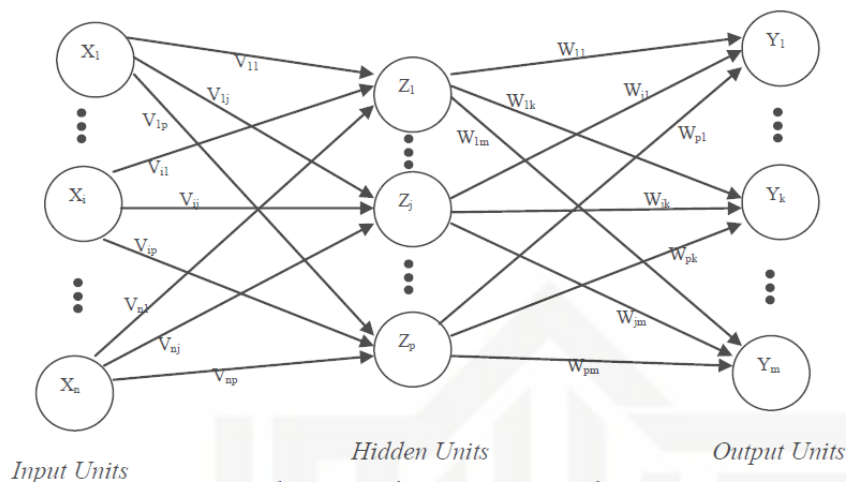
Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 2.2 Jaringan Lapisan Jamak (Ratnaningtyas, 2013)

Keterangan:

$x_{1,}, \dots, x_n$: Nilai *input* ke 1 sampai n

$z_{1,}, \dots, z_p$: Nilai dari *neuron hidden layer*

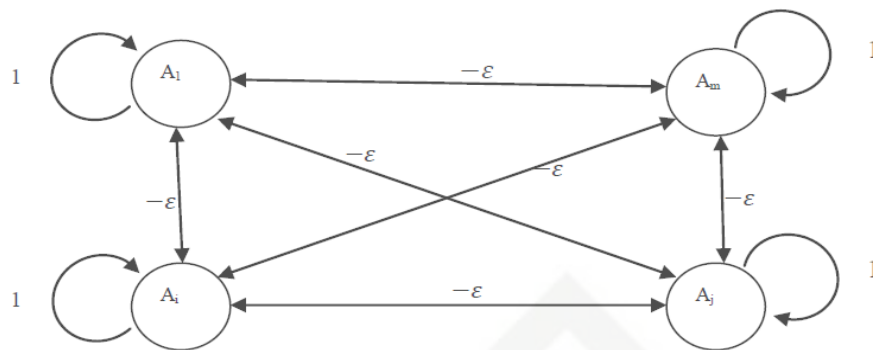
$y_{1,}, \dots, y_m$: Nilai *output* hasil pembangkitan nilai *input* oleh suatu fungsi aktivasi

W_{ij} : Bobot atau nilai

Pada Gambar 2.2 memperlihatkan jaringan dengan n buah unit masukan (x_1, x_2, \dots, x_n), sebuah *layer* tersembunyi yang terdiri dari p buah unit (z_1, z_2, \dots, z_p) dan m buah unit keluaran (y_1, y_2, \dots, y_m). Jaringan *layer* jamak dapat menyelesaikan masalah yang lebih kompleks dibandingkan dengan *layer* tunggal, meskipun kadangkala proses pelatihan lebih kompleks dan lama.

2.1.1.3 Jaringan Lapisan Kompetitif (Competitive Layer Network)

Jaringan ini memiliki bobot yang telah ditentukan dan tidak memiliki proses pelatihan. Pada jaringan ini neuron saling berkompetisi sehingga hanya satu diantaranya yang aktif, prinsip ini disebut *winner-takes-all* atau yang menanglah yang mengambil bagiannya. Nilai bobot setiap neuron untuk dirinya sendiri adalah 1, sedangkan untuk neuron lainnya bernilai random negative. Contoh JST yang menggunakan jaringan dengan lapisan kompetitif adalah LVQ.



Gambar 2.3 Jaringan Lapisan Kompetitif (Sutojo, Mulyanto dan Suhartono, 2011)

2.1.2 Aturan Pembelajaran Jaringan Syaraf Tiruan

Sebelum digunakan, JST dilatih untuk mengenal fungsi pemetaan. Pelatihan merupakan proses belajar JST yang dilakukan dengan menyesuaikan bobot interkoneksi jaringan. Proses belajar JST diklasifikasikan menjadi dua :

2.1.2.1 Belajar Dengan Pengawasan (*Supervised Learning*)

Jaringan memberikan tanggapan dengan mendapatkan target tertentu. Sebelum jaringan mengubah sendiri bobotnya untuk mencapai target, bobot interkoneksi diinisialisasi. Proses belajar JST dengan pengawasan adalah proses belajar dengan memberikan latihan untuk mencapai suatu target keluaran yang ditentukan. JST mendapatkan latihan untuk mengenal pola-pola tertentu. Dengan memberikan target keluaran, perubahan masukan akan diadaptasi oleh keluaran dengan mengubah bobot interkoneksinya mengikuti algoritma belajar yang ditentukan. Set pelatihan dipilih dari fungsi keluaran maksimum setiap keadaan parameter yang diubah. Dengan menginisialisasi bobot tiap sel, JST akan mencari *error* terkecil, sehingga bentuk fungsi keluaran mendekati target yang diinginkan.

Berdasarkan proses belajar yang dilakukan, kita perlu memperhatikan beberapa hal dalam menyusun set pelatihan, yaitu:

1. Pemberian urutan pola yang akan diajarkan
2. Kriteria perhitungan *error*
3. Kriteria proses belajar
4. Jumlah iterasi yang harus dilalui

5. Inisialisasi bobot dan parameter awal

Pelatihan dilakukan dengan memberikan pasangan pola-pola masukan dan keluaran. Salah satu contoh JST yang menggunakan pembelajaran dengan pengawasan adalah jaringan *backpropagation*.

2.1.2.2 Belajar Tanpa Pengawasan (*Unsupervised Learning*)

Pada belajar tanpa pengawasan jaringan tidak mendapatkan target, sehingga JST mengatur bobot interkoneksi sendiri. JST akan mengklasifikasikan contoh pola-pola masukan yang tersedia ke dalam kelompok yang berbeda-beda. Tujuan pembelajaran ini adalah mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu.

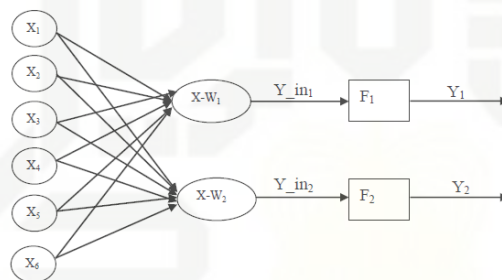
Hal yang ingin dicapai dengan melatih/mengajari jaringan saraf tiruan adalah untuk mencapai keseimbangan antara kemampuan memorisasi dan generalisasi. Yang dimaksud kemampuan memorisasi adalah kemampuan untuk memanggil kembali secara sempurna sebuah pola yang telah dipelajari. Kemampuan generalisasi adalah kemampuan jaringan saraf tiruan untuk menghasilkan respon yang bisa diterima terhadap pola-pola input yang serupa (namun tidak identik) dengan pola-pola yang sebelumnya telah dipelajari. Hal ini sangat bermanfaat bila pada suatu saat ke dalam jaringan saraf tiruan itu diinputkan informasi baru yang belum pernah dipelajari, maka jaringan saraf tiruan itu masih tetap dapat memberikan tanggapan yang baik, memberikan keluaran yang paling mendekati.

2.2 Learning Vector Quantization (LVQ)

Menurut Jang, et al. (1997) LVQ merupakan metode klasifikasi data adaptif berdasarkan pada data pelatihan dengan informasi kelas yang diinginkan. Walaupun merupakan suatu metoda pelatihan *supervised* tetapi LVQ menggunakan teknik data *clustering unsupervised* untuk pra proses set data dan penentuan *cluster center*-nya.

Learning Vector Quantization awalnya diusulkan oleh Kohonen sebagai perbaikan dari *Vector quantization*. *Learning vector quantization* adalah pendekatan yang digunakan untuk pengklasifikasian (Biehl, 2006). Hal ini diterapkan dalam berbagai praktis masalah, termasuk medis dan analisis data.

Learning vector quantization merupakan salah satu jaringan saraf tiruan, dan merupakan versi supervised learning dari algoritma *Kohonen Self-Organizing Map* (SOM). Algoritma *learning vector quantization* bertujuan akhir mencari nilai bobot yang sesuai untuk mengelompokkan vektor-vektor kedalam kelas tujuan yang telah diinisialisasi pada saat pembentukan jaringan *Learning vector quantization*. Pemrosesan yang terjadi pada setiap vektor adalah mencari jarak antara suatu vektor input ke bobot yang bersangkutan (w_1 dan w_2). Untuk lebih jelasnya dapat dilihat pada gambar 2.4 Arsitektur jaringan *Learning vector quantization*.



Gambar 2.4 Arsitektur Jaringan *Learning Vector Quantization*
(Kusumadewi,Hartati, 2006)

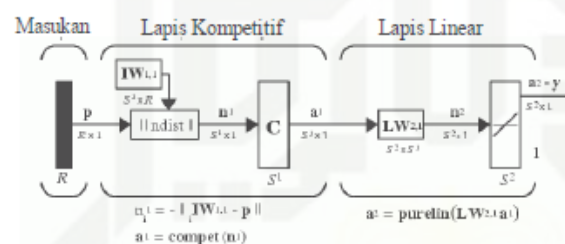
Keterangan:

1. $X_1, X_2 - X_6$ merupakan vektor inputan. Kemudian vektor-vektor input tersebut dihubungkan ke vektor W_1 dan W_2 .
2. W_1 dan W_2 merupakan vektor bobot pertama dan kedua. W_1 merupakan vektor bobot yang menghubungkan setiap neuron pada lapisan input ke neuron pertama pada lapisan output, sedangkan W_2 merupakan vektor bobot yang menghubungkan setiap neuron pada lapisan input ke neuron yang kedua pada lapisan output.
3. F_1 dan F_2 merupakan fungsi aktivasi pertama dan kedua. Fungsi aktivasi F_1 akan memetakan y_{in1} ke $y_1 = 1$ apabila $\|X - w_1\| < \|X - w_2\|$, dan $y_1 = 0$ jika sebaliknya. Demikian pula dengan yang terjadi pada fungsi aktivasi F_2 , akan memetakan y_{in2} ke $y_2 = 2$ apabila $\|X - w_2\| < \|X - w_1\|$, dan $y_2 = 0$ jika sebaliknya.
4. Y_1 dan Y_2 merupakan output pertama dan kedua.

Ada beberapa variasi dari algoritma LVQ yaitu, LVQ1, LVQ2, LVQ2.1 (Kohonen,1990a) dan LVQ3 (Kohonen,1990b). Pada algoritma LVQ1 hanya vektor referensi terdekat (vektor pemenang) dengan vektor masukan yang diperbaharui. Perpindahan vektor tergantung pada apakah vektor referensi memiliki kelas yang sama dengan vektor masukan.

2.2.1 Arsitektur Jaringan LVQ

Berikut adalah arsitektur jaringan dari LVQ:



Gambar 2.5 Arsitektur Jaringan *Larning Vector Quantization* (Wahab, 2010)

Menurut Wahab (2008), LVQ terdiri dari dua lapis yaitu lapis kompetitif dan lapis linier. Lapis pertama yaitu lapis kompetitif atau disebut juga *Self Organizing Map* (SOM), mengklasifikasikan masukan dengan cara kompetitif dimana neuron-neuron berkompetisi dengan algoritma kompetisi yang akan menghasilkan neuron pemenang (*winning neuron*). Lapis kedua yaitu lapis linear mentransformasikan sub-sub kelas keluaran lapis pertama menjadi kelas-kelas yang telah didefinisikan sebelumnya (target). Setiap sub kelas diwakili oleh suatu neuron pada keluaran lapis kompetitif dan setiap kelas diwakili oleh satu neuron pada keluaran lapis linier. Neuron pada keluaran lapis kompetitif biasanya disebut sebagai *hidden* neuron dan neuron pada keluaran lapis linier disebut sebagai output neuron. Sub kelas pada lapis kompetitif adalah hasil kompetisi pada lapis tersebut sedangkan kelas pada lapis linier adalah kelas yang didefinisikan oleh pengguna (target). Pemrosesan yang terjadi pada setiap neuron adalah mencari jarak terdekat antara suatu vektor masukan ke bobot yang bersangkutan. Input akan bersaing untuk dapat masuk ke dalam suatu kelas. Hasil dari lapisan kompetitif ini berupa kelas, yang kemudian akan dihubungkan dengan lapisan

output oleh fungsi aktivasi. Fungsi aktivasi (F) yang digunakan pada arsitektur jaringan LVQ adalah fungsi linier. Tujuannya adalah agar diperoleh keluaran yang sama dengan masukan, sesuai dengan rumus fungsi linear yaitu $y = x$ (Hidayati dan Warsito, 2010).

2.2.2 Algoritma *Learning Vector Quantization* (LVQ 1)

Langkah-langkah algoritma pelatihan LVQ 1 (Kusumadewi, 2004) adalah sebagai berikut:

1. Tetapkan:

- Bobot awal variabel input ke-j menuju kelas (cluster) ke-i.
- Parameter learning rate (α)
- Pengurangan learning rate (Dec α)
- Minimal learning rate yang diperbolehkan (min α)

2. Masukkan:

- Data input: X_{ij}
Dengan $i = 1, 2, \dots, n$; dan $j = 1, 2, \dots, m$.
- Target berupa kelas: T_k ;
Dengan $k = 1, 2, \dots, n$.

3. Tetapkan kondisi awal epoch

4. Kerjakan jika ($\alpha \geq \text{min } \alpha$)

- Epoch = epoch + 1;
- Kerjakan untuk $i = 1$ sampai n
 - Tentukan j sedemikian hingga $\|X_i - W_j\|$
 - Perbaiki W_j dengan ketentuan:
 - Jika $T = C_j$ maka: $W_j = W_j + \alpha (X_i - W_j)$(2.1)
 - Jika $T \neq C_j$ maka: $W_j = W_j - \alpha (X_i - W_j)$(2.2)
- Kurangi nilai α .

(Pengurangan α bisa dilakukan dengan: $\alpha = \alpha - \text{Dec } \alpha$; atau dengan Cara: $\alpha = \alpha * \text{Dec } \alpha$).

Setelah dilakukan pelatihan, akan diperoleh bobot-bobot akhir. Bobot-bobot ini nantinya akan digunakan untuk melakukan simulasi atau pengujian.

Algoritma simulasi (pengujian) :

1. Masukkan data yang akan diuji, misal: X_{ij} dengan $i = 1, 2, \dots, np$; dan $j = 1, 2, \dots, m$.
2. Kerjakan untuk $i = 1$ sampai np
 - a. Tentukan J sedemikian hingga $\|X_i - W_j\|$ minimum; dengan $J = 1, 2, \dots, k$.
 - b. J adalah kelas untuk X_i .

2.2.3 Algoritma *Learning Vector Quantization 2* (LVQ 2)

LVQ2 adalah sebuah algoritma hasil pengembangan dari algoritma LVQ awal (LVQ1) (Budianita, 2013), kondisi dimana kedua vektor akan diperbaharui jika:

1. Unit pemenang dan *runner up* (vektor terdekat kedua) merepresentasikan kelas yang berbeda
2. Vektor masukan mempunyai kelas yang sama dengan *runner up*
3. Jarak antara vektor masukan ke pemenang dan jarak antara vektor masukan ke *runner up* kira-kira sama.

Kondisi ini diperlihatkan di dalam notasi berikut:

X vektor masukan saat ini

Y_c vektor referensi terdekat dengan X

Y_r vektor referensi terdekat berikutnya dengan X (*runner up*)

D_c jarak dari X ke Y_c

D_r jarak dari X ke Y_r

Vektor referensi dapat diperbaharui jika masuk ke dalam daerah yang disebut *window* (ϵ). *Window* yang digunakan untuk memperbaharui vektor referensi didefinisikan sebagai berikut:

Vektor masukan X akan masuk ke dalam *window* bila

$$\frac{d_c}{d_r} > 1 - \epsilon, \quad \frac{d_r}{d_c} < 1 + \epsilon \quad \dots\dots\dots(2.3)$$

dengan nilai ϵ tergantung dari jumlah data pelatihan. Berdasarkan Kohonen (1990a) dalam Fausett (1994) nilai $\epsilon = 0.3$ adalah nilai yang disarankan. Vektor

Y_c dan Y_r akan diperbaharui bila kondisi 1,2 dan 3 terpenuhi. Vektor Y_c dan Y_r diperbaharui dengan menggunakan persamaan :

$$Y_c(t+1) = Y_c(t) - \alpha(t)[X(t) - Y_c(t)] \dots\dots\dots(2.4)$$

$$Y_r(t+1) = Y_r(t) + \alpha(t)[X(t) - Y_r(t)] \dots\dots\dots(2.5)$$

2.2.4 Algoritma *Learning Vector Quantization* 2.1 (LVQ 2.1)

Modifikasi LVQ yang disebut LVQ2.1 (kohonen,1990a) dalam Fausett (1994) mempertimbangkan dua vektor referensi terdekat, yaitu Y_{c1} dan Y_{c2} . Kondisi untuk memperbaharui kedua vektor tersebut adalah apabila salah satu dari vektor tersebut (misal, Y_{c1}) masuk ke dalam kelas yang sama dengan vektor masukan x , sementara vektor lainnya (misal, Y_{c2}) tidak masuk ke dalam kelas yang sama dengan vektor masukan x . Sebagaimana LVQ2, vektor x harus masuk ke dalam window agar bisa terjadi pembaharuan. Window didefinisikan sebagai berikut :

$$\min \left[\frac{dc1}{dc2}, \frac{dc2}{dc1} \right] > 1 - \epsilon$$

$$\max \left[\frac{dc1}{dc2}, \frac{dc2}{dc1} \right] < 1 + \epsilon \dots\dots\dots(2.6)$$

Jika kondisi-kondisi tersebut terpenuhi, maka vektor referensi yang masuk ke dalam kelas yang sama dengan vektor x akan diperbaharui menggunakan persamaan :

$$Y_{c1}(t+1) = Y_{c1}(t) + \alpha(t)[x(t) - Y_{c1}(t)] \dots\dots\dots(2.7)$$

Sedangkan vektor referensi yang tidak masuk ke dalam kelas yang sama dengan vektor x akan diperbaharui menggunakan persamaan :

$$Y_{c2}(t+1) = Y_{c2}(t) - \alpha(t)[x(t) - Y_{c2}(t)] \dots\dots\dots(2.8)$$

2.3 Normalisasi

Dalam perhitungan jarak *euclidean*, atribut berskala panjang dapat mempunyai pengaruh lebih besar daripada atribut berskala pendek. Karena itu, untuk mencegah hal tersebut perlu dilakukan normalisasi terhadap nilai atribut. Normalisasi adalah proses transformasi nilai menjadi skalar antara 0 dan 1 (Teknomo,

2006, dikutip oleh Budianita, 2013) yang bertujuan untuk mendapatkan data dengan ukuran yang lebih kecil yang mewakili data yang asli tanpa kehilangan karakteristik sendirinya (Indrabayu, dkk, 2012)

Rumus dari normalisasi yaitu :

$$X^* = \frac{X - \min(X)}{\max(X) - \min(X)} \dots\dots\dots(2.9)$$

Keterangan:

- X^* : nilai setelah dinormalisasi,
 X : nilai sebelum dinormalisasi,
 $\min(X)$: nilai minimum dari fitur, dan
 $\max(X)$: nilai maksimum dari suatu fitur

Normalisasi merupakan proses penskalaan nilai atribut dari data sehingga bisa jatuh pada *range* tertentu. (Teknomo, 2006,dikutip oleh Budianita, 2013) selanjutnya setelah dinormalisasi, jarak dapat dihitung sebagai variabel kuantitatif. Jarak antara dua benda yang diwakili oleh variabel ordinal dapat ditentukan dengan mengubah skala ordinal ke skala rasio dengan melakukan langkah-langkah:

1. Konversi nilai ordinal menjadi *rank* ($r = 1$ sampai R).
2. Normalisasi peringkat ke nilai 0 sampai 1 menggunakan persamaan :

$$x = \frac{r-1}{R-1} \dots\dots\dots(2.10)$$

3. Jarak dapat dihitung dengan memperlakukan nilai ordinal sebagai variabel kuantitatif (diantaranya dapat menggunakan persamaan jarak *euclidean*, *city blok*, *chebyshev*, *minkowski*, *canberra*, sudut pemisahan, dan koefisien korelasi).

2.4 Pengujian Hasil Akurasi

Tingkat keberhasilan sistem dihitung dari perbandingan jumlah klasifikasi yang sesuai terhadap seluruh data pasien hipertensi atau tekanan darah tinggi yang diujikan sehingga menghasilkan persamaan.

$$\text{Akurasi} = \frac{\sum \text{Pengujian yang bernilai benar}}{\sum \text{Banyak Data Uji}} \times 100\% \dots \dots \dots (2.11)$$

2.5 Tuberculosis (TBC) Paru

Tuberkulosis paru adalah tuberkulosis yang menyerang jaringan paru, tidak termasuk pleura. Menurut Misnadiarly (2006: 11), penyakit TBC merupakan penyakit kronis menahun yang telah lama dikenal oleh masyarakat luas dan ditakuti karena menular. Pada 1882, Robert Kock secara meyakinkan telah dapat memberikan bukti bahwa TBC adalah suatu penyakit infeksi yang disebabkan oleh bakteri yang diberi nama *Mycobacterium tuberculosis*. Bakteri tersebut berwarna merah, berbentuk batang, dan tahan asam disebut Basil Tahan Asam (BTA). Orang yang pertama kali dapat membuktikan bahwa TBC adalah penyakit yang dapat ditularkan adalah Villemin yang hidup pada tahun 1827-1894. Menurut Robbins, sebagaimana dikutip oleh Misnadiarly (2006: 11), TBC adalah penyakit infeksi kronis yang disebabkan oleh *Mycobacterium tuberculosis*, dan biasa terdapat pada paru-paru, tetapi mungkin juga pada organ lain seperti kelenjar getah bening (*nodus lymphaticus*). Bakteri TBC yang masuk ke dalam kelenjar getah bening dapat menimbulkan peradangan pada kelenjar tersebut disertai perubahan struktur jaringan (Ratnaningtyas, 2013).

2.5.1 Penyebab dan Faktor Risiko Tuberkulosis

Penyebab tuberkulosis adalah bakteri yang menyebar di udara melalui semburan air liur dari batuk atau bersin pengidap TB. Nama bakteri TB adalah *Mycobacterium Tuberculosis*. *Mycobacterium Tuberculosis* berbentuk batang lurus atau sedikit melengkung, tidak berspora dan tidak berkapsul. Bakteri ini berukuran lebar 0,3 – 0,6 mm dan panjang 1 – 4 mm. Dinding *M. tuberculosis* sangat kompleks, terdiri dari lapisan lemak cukup tinggi (60%). Penyusun utama dinding sel *M. tuberculosis* ialah asam mikolat, lilin kompleks (complex-waxes), trehalosa dimikolat yang disebut *cord factor*, dan *mycobacterial sulfolipids* yang berperan dalam virulensi. Asam mikolat merupakan asam lemak berantai panjang (C60 – C90) yang dihubungkan dengan arabinogalaktan oleh ikatan glikolipid dan dengan peptidoglikan oleh jembatan fosfodiester. Unsur lain yang terdapat pada dinding sel bakteri tersebut adalah polisakarida seperti arabinogalaktan dan

arabinomanan. Struktur dinding sel yang kompleks tersebut menyebabkan bakteri *M. tuberculosis* bersifat tahan asam, yaitu apabila sekali diwarnai akan tetap tahan terhadap upaya penghilangan zat warna tersebut dengan larutan asam – alkohol.

Beberapa faktor penyebab TBC (Depkes RI, 1997) adalah sebagai berikut:

1. Faktor sosial ekonomi

Faktor sosial ekonomi ini sangat erat kaitannya dengan kondisi rumah, kepadatan hunian, lingkungan perumahan serta sanitasi. Semua faktor tersebut dapat memudahkan penularan TBC, karena pendapatan yang kecil membuat orang tidak dapat hidup layak, yang memenuhi syarat-syarat kesehatan

2. Status Gizi

Kekurangan kalori, protein, vitamin, zat besi, dan lain-lain (malnutrisi) akan mempengaruhi daya tahan tubuh seseorang, sehingga rentan terhadap berbagai penyakit, termasuk tbc paru. Keadaan ini merupakan faktor penting yang berpengaruh di negara miskin, baik pada orang dewasa maupun anak-anak.

3. Umur

Penyakit tbc paru paling sering ditemukan pada usia muda atau usia produktif yaitu 15-30 tahun. Dewasa ini, dengan terjadinya transisi demografi menyebabkan usia harapan hidup lansia menjadi lebih tinggi. Pada usia lanjut, lebih dari usia 55 tahun system imunologis seseorang menurun, sehingga sangat rentan terhadap berbagai penyakit, termasuk tuberkolosis paru.

4. Jenis Kelamin

Menurut WHO, sedikitnya dalam periode setahun ada sekitar 1 juta perempuan yang meninggal akibat tbc paru. Dari fakta ini dapat disimpulkan bahwa kaum perempuan lebih rentan terhadap kematian akibat serangan tbc paru dibandingkan akibat proses kehamilan dan persalinan. Pada laki-laki, penyakit ini lebih tinggi, karena rokok dan minuman alkohol dapat menurunkan system pertahanan tubuh. Sehingga wajar jika perokok dan peminum alkohol sering disebut sebagai agen dari penyakit tuberkolosis paru.

Berikut ini adalah beberapa kelompok orang yang memiliki risiko lebih tinggi tertular TB:

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

1. Orang dengan sistem kekebalan tubuh yang menurun, misalnya pengidap HIV/AIDS, diabetes atau orang yang sedang menjalani kemoterapi.
2. Orang yang mengalami malnutrisi atau kekurangan gizi.
3. Pecandu narkoba.
4. Para perokok.
5. Para petugas medis yang sering berhubungan dengan pengidap TB.

2.5.3 Klasifikasi Tingkat Keparahan TBC

Klasifikasi TBC berdasarkan tingkat keparahan penyakit (Depkes RI, 2006) yaitu:

1. TBC paru ringan yang ditandai dengan gejala:
 - Batuk (kadang-kadang atau tidak sama sekali)
 - Nafsu makan menurun
 - Demam (demam biasa atau demam malam hari)
2. TBC paru berat yang ditandai dengan gejala:
 - Batuk yang berlangsung selama 3 minggu atau lebih
 - Batuk berdahak dimana dahak yang dihasilkan berwarna putih atau kekuningan
 - Batuk berdarah
 - Demam dimalam hari
 - Nyeri pada dada (sering, jarang, atau tidak pernah)
 - Sesak napas (berat dan berulang, sedang, atau tidak sesak)
 - Kehilangan nafsu makan
 - Berat badan berkurang
 - Hasil pemeriksaan BTA positif.
 - Foto toraks memperlihatkan gambaran kerusakan paru yang lebih luas (Lesi luas)

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.5.3 Pencegahan Penyakit TBC Paru

Banyak hal yang bisa dilakukan mencegah terjangkitnya TBC paru. Pencegahan berikut dapat dilakukan oleh penderita, masyarakat, maupun petugas kesehatan.

1. Bagi penderita, pencegahan penularan dapat dilakukan dengan menutup mulut saat batuk, dan membuang dahak tidak di sembarang tempat.
2. Bagi masyarakat, pencegahan peneluran dapat dilakukan dengan meningkatkan ketahanan terhadap bayi yaitu dengan memberikan vaksin BCG.
3. Bagi petugas kesehatan, pencegahan dapat dilakukan dengan member penyuluhan tentang penyakit TBC yang meliputi gejala, bahaya, dan akibat yang ditimbulkannya terhadap kahidupan masyarakat pada umumnya.
4. Petugas kesehatan juga harus segera melakukan pengisolasian dan pemeriksaan terhadap orang-orang yang terinfeksi, atau dengan memberikan pengobatan khusus kepada penderita TBC ini. Pengobatan dengan cara menginap di rumah sakit hanya dilakukan bagi penderita dengan kategori berat dan memerlukan pengembangan program pengobatannya, sehingga tidak dikehendaki pengobatan jalan.
5. Pencegahan penularan juga dapat dicegah dengan melaksanakan desinfeksi, seperti cuci tangan, kebersihan rumah yang ketat, perhatian khusus terhadap muntahan atau ludah anggota keluarga yang terjangkit penyakit ini (piring, tempat tidur, pakaian), dan menyediakan ventilasi rumah dan sinar matahari yang cukup.
6. Melakukan imunisasi orang-orang yang melakukan kontak langsung dengan penderita, seperti keluarga, perawat, dokter, petugas kesehatan, dan orang lain yang terindikasi, dengan vaksin BCG dan tindak lanjut bagi yang positif tertular.
7. Melakukan penyelidikan terhadap orang-orang kontak. Perlu dilakukan Tes Tuberculin bagi seluruh anggota keluarga. Apabila cara ini

menunjukkan hasil negatif, perlu diulang pemeriksaan tiap bulan selama 3 bulan dan perlu penyelidikan intensif

8. Dilakukan Pengobatan khusus. Penderita dengan TBC aktif perlu pengobatan yang tepat, yaitu obat-obat kombinasi yang telah ditetapkan oleh dokter untuk diminum dengan tekun dan teratur selama 6-12 bulan. Perlu diwaspadai adanya kebal terhadap obat-obat, dengan pemeriksaan penyelidikan oleh dokter

2.6 Penelitian Terkait

Tabel 2.1 berikut merupakan penelitian yang pernah dilakukan sebelumnya tentang penyakit TBC Paru

Tabel 2.1 Penelitian Terkait Tentang Penyakit TBC Paru

No	Peneliti	Universitas	Judul	Metode	Akurasi
1	Akbar Wira Pradana	Universitas Pamulang	Penerapan Metode Jaringan Syaraf Tiruan Untuk Mendeteksi Penyakit Tuberkolosis Paru Pada Manusia	<i>Backpropagation</i>	97%
2	Enny Ide Sela, Sri Hartati	STMIK, Universitas Gajah Mada	Pengenalan Jenis Penyakit THT Menggunakan Jaringan Learning Vector Quantization	LVQ	94%
3	Bambang Yuwono, Heru Cahya Rustamaji, Usamah Dani (2011)	UPN "Veteran" Yogyakarta	Diagnosa Gangguan Saluran Pernafasan Menggunakan Jaringan Syaraf Tiruan <i>Backpropagation</i>	<i>Backpropagation</i>	90%
4	Fajar Rohman Hariri	STMIK AMIKOM	Implementasi <i>Learning Vector Quantization</i> Untuk Diagnosa Penyakit Diabetes	LVQ	86%

	(2013)		Mellitus		
5	Novi Indah Pradasar,F .Trias Pontia &Dedi Triyanto (2013)	Universitas Tanjung Pura	Aplikasi Jaringan Syaraf Tiruan Untuk Memprediksi Penyakit Saluran Pernafasan Dengan Metode Backpropagation	Backpropagation	91,66%
6	Hessy Gustina Suri (2016)	UIN SUSKA RIAU	Diagnosa Penyakit Stroke Menggunakan Metode Learning Vector Quantization 2.1 Berdasarkan Nilai Ketidakpastian Sistem Pakar	LVQ 2.1	100%

2.7.1 Penerapan Metode Jaringan Syaraf Tiruan Untuk Mendeteksi Penyakit Tuberkolosis Paru Pada Manusia

Penelitian untuk mendeteksi penyakit tuberkolosis paru pada manusia. Parameter yang digunakan berjumlah 8, yaitu kategori batuk, sesak nafas, nyeri dada demam pada sore atau malam hari, penurunan nafsu makan, badan lemah, hasil lab(BTA), dan riwayat tbc. Adapun untuk pembagian kelasnya menggunakan 2 kelas, yaitu suspect TB Paru dan Negatif TB Paru. Tingkat akurasi yang diperoleh menggunakan metode ini adalah 97% (Akbar Wira Pradana, 2013).

2.7.2 Pengenalan Jenis Penyakit THT Menggunakan Jaringan Learning Vector Quantization

Penelitian tentang pengenalan jenis penyakit THT. Parameter yang digunakan berjumlah 13, yaitu hidung tersumbat, pilek, bersin, kelar sekret, hidung gatal, batuk, alergi nyeri kepala/pusing, demam, keluar darah, lemas, nyeri tekan, dan tidak enak badan. Adapun untuk pembagian kelasnya menggunakan 3

kelas, yaitu penyakit EP, Penyakit RK, dan penyakit SI.PJK dan bukan PJK, sedangkan jumlah data yang digunakan sebanyak 40 data.Tingkat akurasi yang diperoleh menggunakan metode LVQ adalah 94% (Enny Ide Sela, Sri Hartati).

2.7.3. Diagnosa Gangguan Saluran Pernafasan Menggunakan

Jaringan Syaraf Tiruan *Backpropagation*

Penelitian tentang gangguan saluran pernafasan manusia menggunakanmetode*backpropagation*.Parameter yang digunakan berjumlah 10 yaitu: sesak napas, batuk berdahak, nyeri dada, panas, gangguan penglihatan, pembengkakan pergelangan kaki dan kaki, benjolan di ketiak, denyut jantung cepat, kekakuan, tekanan darah rendah. Tingkat akurasi yang diperoleh menggunakan metode ini adalah 90% (Bambang Yuwono, Heru Cahya Rustamaji, Usamah Dani, 2011)

2.7.4 Implementasi *Learning Vector Quantization* Untuk Diagnosa Penyakit

Diabetes Mellitus

Penelitian tentang penyakit diabetes melitus menggunakan metode *backpropagation*. Parameter berjumlah 8 yaitu: glukosa darah puasa, kadar glukosa plasma puasa, kadar glukosa plasma tidur, kadar insulin, kadar HbA1c, kadar kolesterol HDL, kada trigliserida dan umur. Tingkat akurasi yang diperoleh menggunakan metode ini adalah 86% (Fajar Rohman Hariri, 2013)

2.7.5 Aplikasi Jaringan Syaraf Tiruan Untuk Memprediksi Penyakit Saluran Pernafasan Dengan Metode *Backpropagation*

Penelitian tentang penyakit pada saluran pernafasan menggunakan metode *backpropagation*. Parameter berjumlah 12 yaitu sesak nafas,batuk,nyeri dada,demam,tekanan darah rendah,mual dan muntah,menggigil,hidung tersumbat,sakit kepala,berat badan turun,berkeringat dingin,kekakuan sendi.Tingkat akurasi yang diperoleh menggunakan metode ini adalah 91,66% (Novi Indah Pradasar, F.Trias Pontia &Dedi Triyanto, 2013).

2.7.6 Diagnosa Penyakit Stroke Menggunakan Metode *Learning Vector Quantization* 2.1 Berdasarkan Nilai Ketidakpastian Sistem Pakar

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penelitian mengenai sebuah aplikasi yang bisa mengenali atau mendiagnosa penyakit stroke. Inputan diambil berdasarkan gejala-gejala stroke pada sistem pakar. Sedangkan outputnya terdiri dari enam kelas yaitu TIA (*Transient Ischemic Attack*), RIND (*Reversible Ischemic Neurologic Deficits*), Stroke Progresif, Stroke Komplet, Hemoragik Interaserebral dan Hemoragik Subaraknoid. Pengujian pada penelitian ini dilakukan dengan dua cara yaitu pengujian berdasarkan LVQ2.1 dengan persentase yang diperoleh sebesar 100% dan pengujian berdasarkan pohon inferensi dengan persentase 80%.